

IMAGINE MY BUSINESS

Arezzo, 14 Settembre 2023



Shopify Editions
Summer '23

/DEV MODE



www.shopify.com/editions



Beniamino Colombo

Senior Solutions Engineer
Shopify Plus



Sintra + Bitforce



Tommaso Rossi

CEO & Front end CTO
Bitforce



Francesco Todino

Front end Developer
Bitforce



Mattia Rosi

Front end Developer
Bitforce



Riccardo Gai

Front end Developer
Bitforce



Luca Innocenti

Software Architect & Product
Sintra



Fabiola Campioli

Front end Developer
Sintra

Shopify: Summer '23 Edition

AGENDA WORKSHOP DEV

1.

Headless con Shopify [10:00 - 11:40]

- Intro su headless
- Let's go headless
- Up and running with Hydrogen
- Una product page headless
- Metaobjects e deploy con Oxygen
- Q&A
- Pausa 15 min

2.

Integrazioni [11:40 - 12:25]

- Dati e contenuti in contesto enterprise
- I flussi dati e gli interlocutori
- Casi reali e soluzioni tecniche
- Sistema Ipaas per le integrazioni
- Q&A
- Pausa 5 min

3.

Nuove possibilità per Shopify B2B [12:25 - 13:00]

- Layout B2B per clienti B2B
- Metaobjects e landing pages
- Estendere l'esperienza di checkout
- Q&A





Headless con Shopify



Headless

Full Flexibility

Buy Shopify

or anywhere in between

Build with Shopify



Native functionality



Rapid deployment



Leverage pre-built



Composable / Headless



Advance customisation



Partner Ecosystem

Shopify-managed What you manage

Online Store

Custom Storefronts

Stack
summary

Web-based storefront
built using Liquid,
hosted by Shopify

Web-based storefront
built using Hydrogen,
hosted by Shopify

Web-based storefront
built using JavaScript,
self-hosted

App-based storefront

Example

NETFLIX

DRAKE RELATED

THE ORIGINAL
OAT
LY!

FASHIONNOVA

What hosts
the storefront

 **shopify**
Fast, secure, global hosting

 **shopifyOXYGEN**
Fast, secure, global hosting

  **Vercel**
Bring your own hosting



What
developers use
to build the
storefront




HYDROGEN



Developer Tools
Mobile SDKs (iOS & Android),
external tools

How Shopify
data is
accessed

Liquid rendering engine
(server-side rendering)

Storefront Edge API

Where you
manage your
online
presence

Backoffice



Let's go Headless!



Di cosa parleremo

- Scopriamo la tecnologia headless
- Stack Shopify headless
- Intro su Hydrogen - Oxygen - Storefront API
- Nei prossimi speech
 - Up and running with Hydrogen
 - Una product page headless
 - Metaobjects e deploy con Oxygen





Tommaso Rossi

CEO & Front end CTO @bitforce
sintra partner

Cosa facciamo in Bitforce

- Sviluppiamo soluzioni headless su stack javascript / node
- Scriviamo software custom:
 - Web App / PWA / App ibrido native
- Consulenze su incremento di performance tecniche di siti web (PSI / WCV)
- Con i nostri partner Sintra e Shopify realizziamo headless e-commerce



Headless for dummies

1. Si disaccoppia la UI dai dati
2. Per mettere a schermo l'html serve un framework
Front End
3. Per ottenere i dati serve un servizio di API
4. Si sviluppa il codice
5. Si deploia in prod!



Vantaggi headless

- Si scelgono i vari componenti dello stack:
 - Framework di rendering UI
 - Sorgente di dati - multiple sorgenti
 - CDN per immagini / video / assets
 - Servizi di hosting
- Si rimuovono molte limitazioni:
 - Personalizzazioni del design illimitate
 - Non ci sono limiti requisiti funzionali
 - La UX viene portata ai massimi livelli con framework front end.
- Performance al top, grazie a SSR ed al fatto che non si installano apps
- 🔥🔥🔥 Si migliorano le conversioni 💰💰💰



Le possibilità headless sono endless

- UI custom (nessun vincolo dato dai temi)
 - Landing page custom
 - Sezioni per eventi promozionali / a tempo
 - Siti tematici legati a campagne o brand
- PWA: installabili, / notifiche push native / offline
- Multi istanza (Si possono creare più siti collegati al solito commerce)
 - Apps per chioschi ed experience in store
- UX avanzate
 - Gamification, virtual try on
 - Configuratori di prodotto
 - Back office utente avanzati e personalizzati
- Integrazioni di qualsiasi third party in maniera controllata e performante



Case TooA

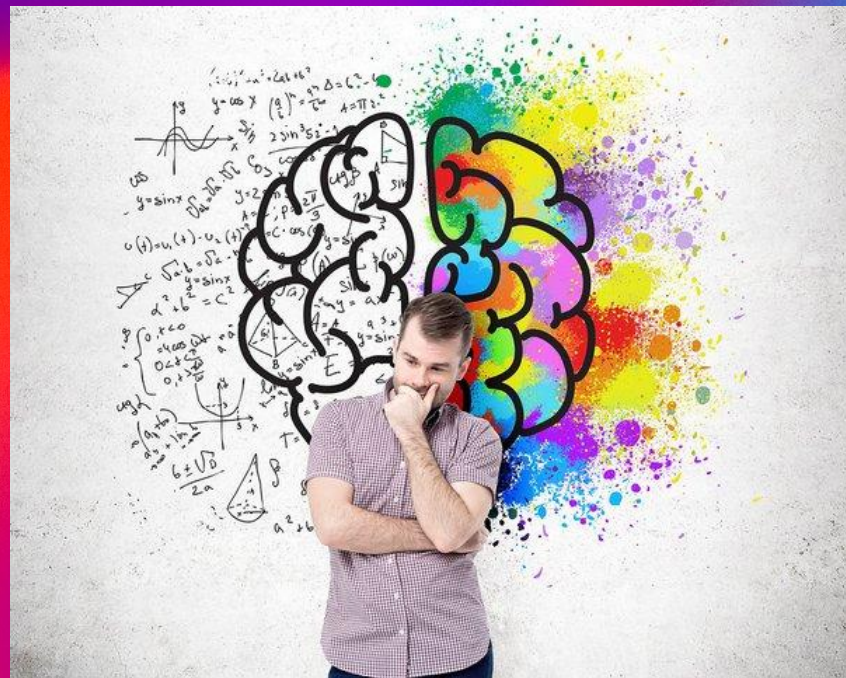
- Sviluppo app
 - Hydrogen / Oxygen / Storefront API
- Layer nativo
 - Capacitor
 - Bluetooth API
 - Scandit API
- Stores
 - Play store
 - Apple store



Complessità headless

- Vanno scelte le giuste componenti del sistema da far dialogare tra loro
- L'esperienza del dev aiuta molto
- Bisogna lavorare bene per mantenere le performances alte

Vedremo come Shopify ci aiuta in questo



Stack **Shopify** headless

- Linguaggio di programmazione
 - **Javascript + Typescript**
- Framework di rendering
 - **React** / Vue / Angular ...
- Framework per rendering server side
 - **Remix** / Next / Nuxt / Gatsby ...
- UI/UX Components
 - **Hydrogen react** / Material / Onsen
- Framework per la gestione degli stili
 - **Tailwind** / Bootstrap / Material ...
- Set di API
 - **GraphQL (storefront API)** / Rest
- Assets (images / video)
 - **Shopify CDN** / imglx / imagekit
- Auth / CIAM
 - **Multipass - Customer API** / Auth0 ...
- Deploy
 - **Oxygen** / AWS / Google cloud / Vercel



Survey live in streaming con hydrogen

- React / Remix / Hydrogen / Oxygen
- Survey.js (hosting per immagini)
- Firebase - Firestore (DB realtime nosql)
- Chart.js
- Tailwind
- browser-image-compression



<https://shorturl.at/ktQR9>

Risorse

- **Remix:** <https://remix.run/>
- **Hydrogen:** <https://hydrogen.shopify.dev/>
- **Storefront API:** <https://shopify.dev/docs/api/storefront>
- **Admin API:** <https://shopify.dev/docs/api/admin>
- **Tailwind:** <https://tailwindcss.com/>
- **GraphQL:** <https://graphql.org/>
- **Youtube channel:**
<https://www.youtube.com/playlist?list=PLvQF73bM4-5XoWU-OnlYsul5wDWWIObzh>
- **Discord:** <https://discord.gg/shopifydevs>



GRAZIE!

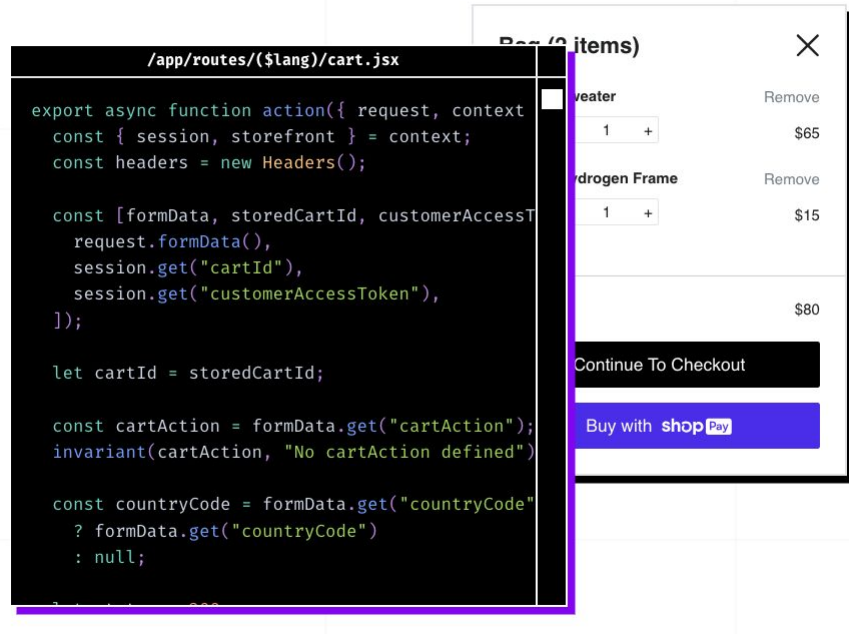


t.rossi@bitforce.app

React + Remix + Hydrogen

- React (framework di base)
 - Suspense, streaming
- Remix (Acquistato da Shopify)
 - Sistema di rotte nestate
 - SSR con workers o node
 - Optimistic updates /Pagination / scroll restore
- Hydrogen (Componenti ed hooks opinionati)
 - Context per gestione utente, carrello
 - Componenti UI es gestione Immagini
 - Gestione cache su pagine e query
 - Utility per accedere alle Store front API

Hydrogen



- Remix approccio minimalistico basato sugli standard GET/POST-form

Remix / SSR

- SSR: Primo rendering server side, poi si passa al client in modalità SPA
- Client side handling
 - Non c'è bisogno di scaricare gli assets
 - Gli assets non devono essere parsati ed eseguiti
 - Si renderizza solo quello che cambia e non tutta la pagina
 - I dati fetchati sono solo quelli che servono

Remix

```
export async function loader({ request }) {
  return getProjects();
}

export async function action({ request }) {
  const form = await request.formData();
  return createProject({ title: form.get("title") });
}

export default function Projects() {
  const projects = useLoaderData();
  const { state } = useNavigation();
  const busy = state === "submitting";

  return (
    <div>
      {projects.map((project) => (
        <Link to={project.slug}>{project.title}</Link>
      ))}

      <Form method="post">
        <input name="title" />
        <button type="submit" disabled={busy}>
          {busy ? "Creating..." : "Create New Project"}
        </button>
      </Form>
    </div>
  );
}
```


- Remix approccio minimalistico basato sugli standard GET/POST-form

Remix / SSR

- SSR: Primo rendering server side, poi si passa al client in modalità SPA
- Client side handling
 - Non c'è bisogno di scaricare gli assets
 - Gli assets non devono essere parsati ed eseguiti
 - Si renderizza solo quello che cambia e non tutta la pagina
 - I dati fetchati sono solo quelli che servono

Remix

```
export async function loader({ request }) {
  return getProjects();
}

export async function action({ request }) {
  const form = await request.formData();
  return createProject({ title: form.get("title") });
}

export default function Projects() {
  const projects = useLoaderData();
  const { state } = useNavigation();
  const busy = state === "submitting";

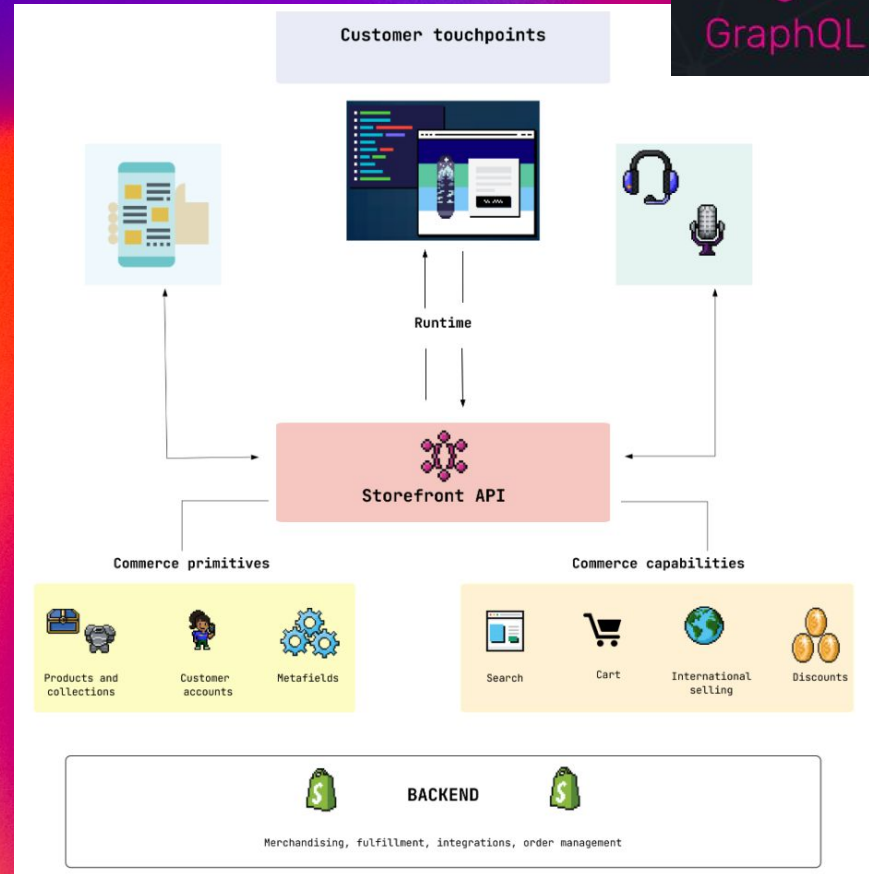
  return (
    <div>
      {projects.map((project) => (
        <Link to={project.slug}>{project.title}</Link>
      ))}

      <Form method="post">
        <input name="title" />
        <button type="submit" disabled={busy}>
          {busy ? "Creating..." : "Create New Project"}
        </button>
      </Form>
    </div>
  );
}
```

Set di API GraphQL per ottenere tutto quello che serve per renderizzare un e-commerce headless

Storefront API

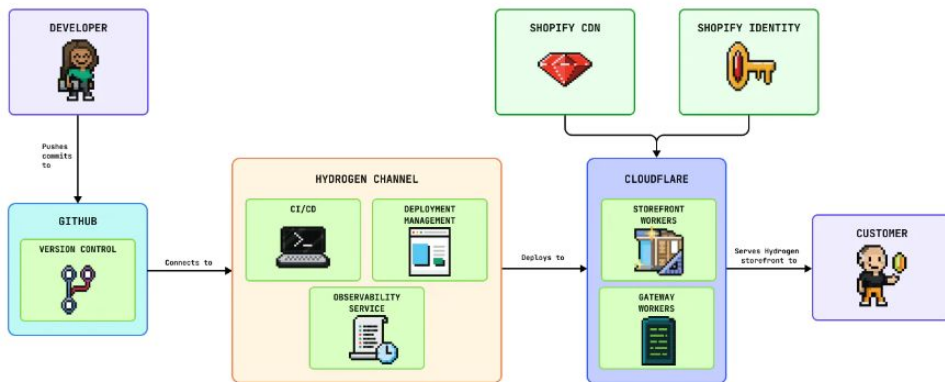
- Prodotti e varianti
- Filtri avanzati - Variant / metafields + Facets
- Pagine e blog
- Accesso alla struttura del menu
- Customer data - profile orders
- Supporto per multi lingua e multi currency
- 🔥 Metaobjects - Gestione contenuti custom
- Unstable: Customer API
- WIP: B2B
- + Admin API per accesso a dati privati dello store



Hosting platform gestita da Shopify

Oxygen

- Solita infrastruttura di Shopify liquid (cloudflare)
- Setup 0 config - Autoscaling / resilience
- Deploy su workers edge distribuiti
- CI/CD collegata a repo GitHub
- Fast rollback
- Multiple environments
- 🔥 Front end (SSR) ed API stanno nel solito network questo permette di fare query molto veloci
- + E' possibile deployare anche in altri servizi





Up and running with Hydrogen





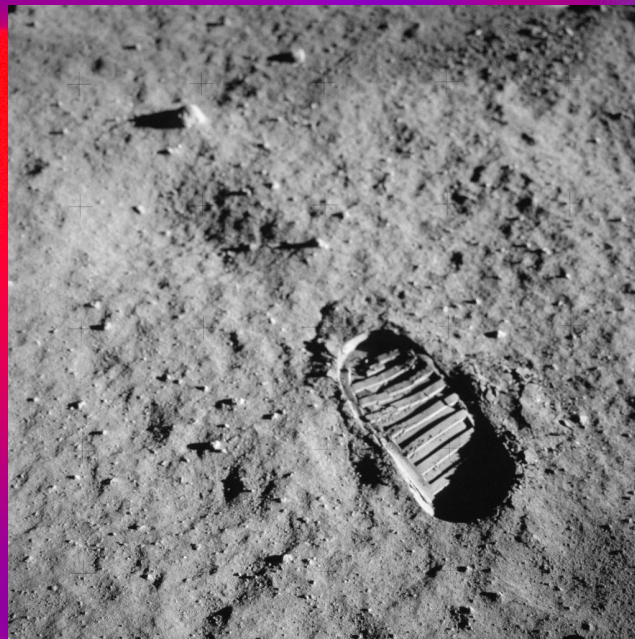
Francesco Todino

Frontend developer

Bitforce

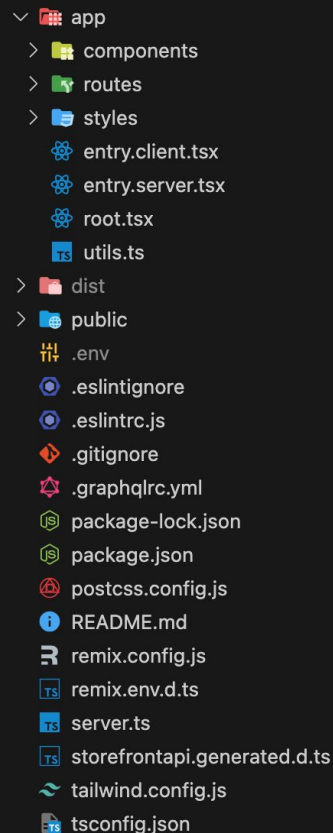
I primi passi

- Ready - i requisiti
 - Node.js, npm
- Set - la creazione
 - `npm create @shopify/hydrogen@latest`
- Go - l'inizio
 - `npm run dev`



La prima esplorazione

- I file del client e del server si troveranno all'interno di **/app**
- La cartella **/dist** sarà composta dalla build del progetto per il deploy
- All'interno di **/public** ci saranno tutti gli assets pubblici
- Nella root / troviamo tutti i file di configurazione



```

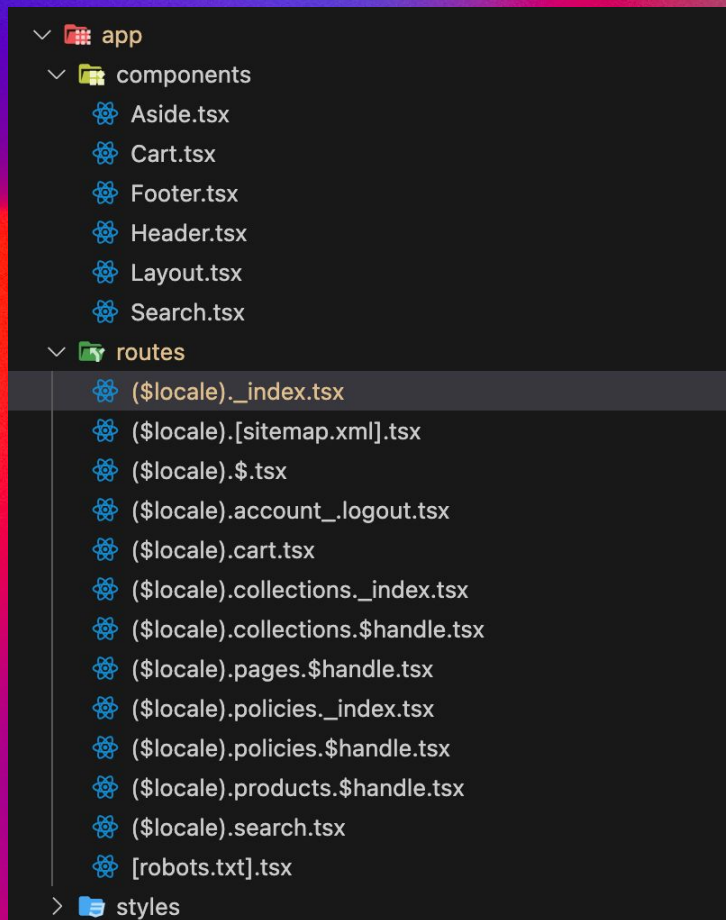
  app
  ├── components
  ├── routes
  ├── styles
  │   ├── entry.client.tsx
  │   ├── entry.server.tsx
  │   ├── root.tsx
  │   └── utils.ts
  ├── dist
  ├── public
  ├── .env
  ├── .eslintignore
  ├── .eslintrc.js
  ├── .gitignore
  ├── .graphqlrc.yml
  ├── package-lock.json
  ├── package.json
  ├── postcss.config.js
  ├── README.md
  ├── remix.config.js
  ├── remix.env.d.ts
  ├── server.ts
  ├── storefrontapi.generated.d.ts
  ├── tailwind.config.js
  └── tsconfig.json

```

A screenshot of a file explorer showing the project structure. The root directory is expanded, showing subdirectories and files. The 'app' directory is expanded, showing 'components', 'routes', 'styles' (containing 'entry.client.tsx', 'entry.server.tsx', 'root.tsx', and 'utils.ts'), 'dist', and 'public'. Other files in the root include '.env', '.eslintignore', '.eslintrc.js', '.gitignore', '.graphqlrc.yml', 'package-lock.json', 'package.json', 'postcss.config.js', 'README.md', 'remix.config.js', 'remix.env.d.ts', 'server.ts', 'storefrontapi.generated.d.ts', 'tailwind.config.js', and 'tsconfig.json'.

Gestione delle rotte

- La rotta di una pagina è determinata dalla sua posizione nella cartella `/routes` e dal suo nome
- `{$locale}` è la variabile che definisce la lingua dello shop
- La pagina di `_index` ha “_” per indicare la mancanza di parametri
- La variabile `$handle` definisce l’handle dell’oggetto che viene passato alla pagina scelta



II Layout

- La pagina **root** definisce la struttura di base della pagina html dello shop (come il vecchio index.html)
- La parte centrale della root viene racchiusa all'interno di **Layout**
- In Layout viene definita la struttura che caratterizza ogni pagina dello shop, rappresentata da **{children}**

```
export default function App() {
  const nonce = useNonce();
  const data = useLoaderData<typeof loader>();

  return (
    <html lang="en">
      <head>
        <meta charset="utf-8" />
        <meta name="viewport" content="width=device-width,initial-scale=1" />
        <Meta />
        <Links />
      </head>
      <body>
        <Layout {...data}>
          <Outlet />
        </Layout>
        <ScrollRestoration nonce={nonce} />
        <Scripts nonce={nonce} />
        <LiveReload nonce={nonce} />
      </body>
    </html>
  );
}
```

```
export function Layout({
  cart,
  children = null,
  footer,
  header,
  isLoggedIn,
}: LayoutProps) {
  return (
    <>
      <CartAside cart={cart} />
      <SearchAside />
      <MobileMenuAside menu={header.menu} />
      <Header header={header} cart={cart} isLoggedIn={isLoggedIn} />
      <main>{children}</main>
      <Suspense>
        <Await resolve={footer}>
          {(footer) => <Footer menu={footer.menu} />}
        </Await>
      </Suspense>
    </>
  );
}
```

Comunicare con lo shop

- Per ottenere i dati dal nostro shop faremo uso di query **graphql** tramite le **Storefront API**
- Possiamo provare queste API direttamente sul nostro shop localhost/graphiql
- Shopify offre anche un endpoint per testare le API senza shop: mock.shop

{ } GraphQL Query

Format >

Reset ↺



```
1 {
2   product(id: "gid://shopify/Product/7982905098262") {
3     id
4     title
5     description
6     featuredImage {
7       id
8       url
9     }
10  }
11 }
12
```

{ } Response

JSON



```
1 {
2   "data": {
3     "product": {
4       "id": "gid://shopify/Product/7982905098262",
5       "title": "Men's Crewneck",
6       "description": "This high-quality crewneck is perfect for your everyday
look. Made with 100% cotton, it's soft, comfortable, and undeniably stylish.
Full sleeved for a classic look and effortlessly versatile, this cotton crewneck
is a must-have in any wardrobe.",
7       "featuredImage": {
8         "id": "gid://shopify/ProductImage/39774600855574",
9         "url": "
https://cdn.shopify.com/s/files/1/0688/1755/1382/products/GreenMenscrew01.jpg?v=1675454919"
10      }
11    }
12  }
13 }
```

Cosa abbiamo visto

- Come avviare un nuovo progetto Hydrogen
- La configurazione delle API dello shop
- Come è strutturato un progetto
- Il sistema di rotte
- Il layout di una pagina
- Come comunicare con il BE



Risorse

- **Node:** <https://nodejs.org/en>
- **Begin development:**
<https://shopify.dev/docs/custom-storefronts/hydrogen/building/begin-development>
- **Mock shop:** <https://mock.shop/>
- **Demo store:** <https://demostore.mock.shop/>



GRAZIE!





Una product page headless





Mattia Rosi

Frontend developer

Bitforce

Product page

- Due react components principali
 - ProductImage
 - ProductMain
- Load data server side (loader function)
 - Query per recupero prodotto, variante selezionata e lista varianti
 - Uso di defer

```
export default function Product() {  
  const {product, variants} = useLoaderData<typeof loader>();  
  const {selectedVariant} = product;  
  return (  
    <div className="product">  
      <ProductImage image={selectedVariant?.image} />  
      <ProductMain  
        selectedVariant={selectedVariant}  
        product={product}  
        variants={variants}  
      />  
    </div>  
  );  
}
```



Hoodie

\$68.00

Size

☐ Small

☐ Medium

☐ Large

☐ X-Large

Color

☐ Green

☐ Olive

☐ Ocean

☐ Purple

☐ Red

Add to cart

Description

This hoodie is the perfect choice for comfort and

Product Image

- Componente nativo hydrogen
 - Responsive by default
 - Aspect ratio
 - Crop
 - Storefront image source
 - Cdn shopify
 - Sizes (native html image attribute)
 - Srcset automatico e configurabile

```
function ProductImage({image}: {image: ProductVariantFragment['image']}) {  
  if (!image) {  
    return <div className="product-image" />;  
  }  
  return (  
    <div className="product-image">  
      <Image  
        alt={image.altText || 'Product Image'}  
        aspectRatio="1/1"  
        data={image}  
        key={image.id}  
        sizes="(min-width: 45em) 50vw, 100vw"  
      />  
    </div>  
  );  
}
```

```
;
}) {
  const {title, descriptionHtml} = product;
  return (
    <div className="product-main">
      <h1>{title}</h1>
      <ProductPrice selectedVariant={selectedVariant} />
      <br />
      <Suspense
        fallback={
          <ProductForm
            product={product}

```

Hoodie

\$68.00

Size

☐ Small ☐ Medium ☐ Large ☐ X-Large

Color

☐ Green ☐ Olive ☐ Ocean ☐ Purple ☐ Red

Add to cart

Description

This hoodie is the perfect choice for comfort and warmth. Meticulously crafted from 100% cotton, the hoodie features a soft, plush fleece interior and a unisex sizing design. Soft and lightweight, it's sure to be your go-to for chilly days.

Variants

- Product form
 - Variant selector (hydrogen)
 - Product options
 - Add to cart button
 - Cart Form (hydrogen)

```
function ProductForm({
  product,
  selectedVariant,
  variants,
}): {
  product: ProductFragment;
  selectedVariant: ProductFragment['selectedVariant'];
  variants: Array<ProductVariantFragment>;
} {
  return (
    <div className="product-form">
      <VariantSelector
        handle={product.handle}
        options={product.options}
        variants={variants}
      >
        {({option}) => <ProductOptions key={option.name} option={option} />}
      </VariantSelector>
    </div>
  );
}
```

Size

Small

Medium

Large

X-Large

Color

Green

Olive

Ocean

Purple

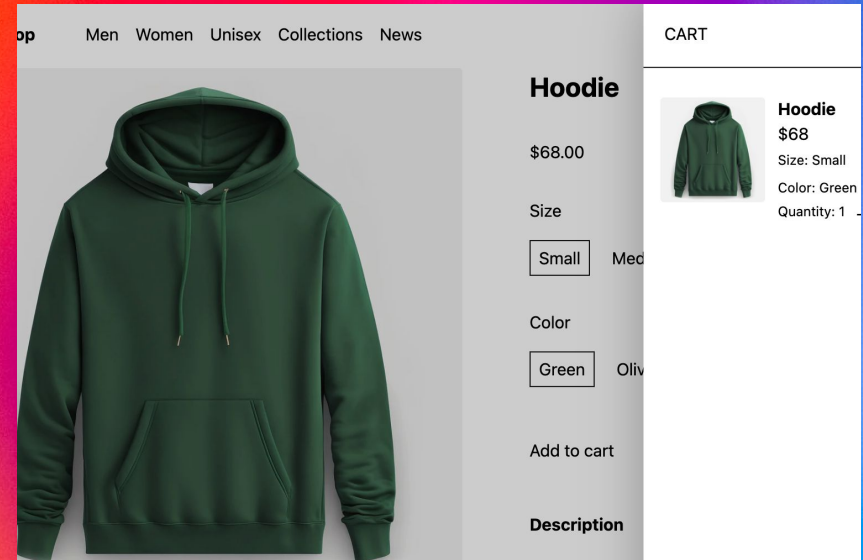
Red

Add to cart

Cart

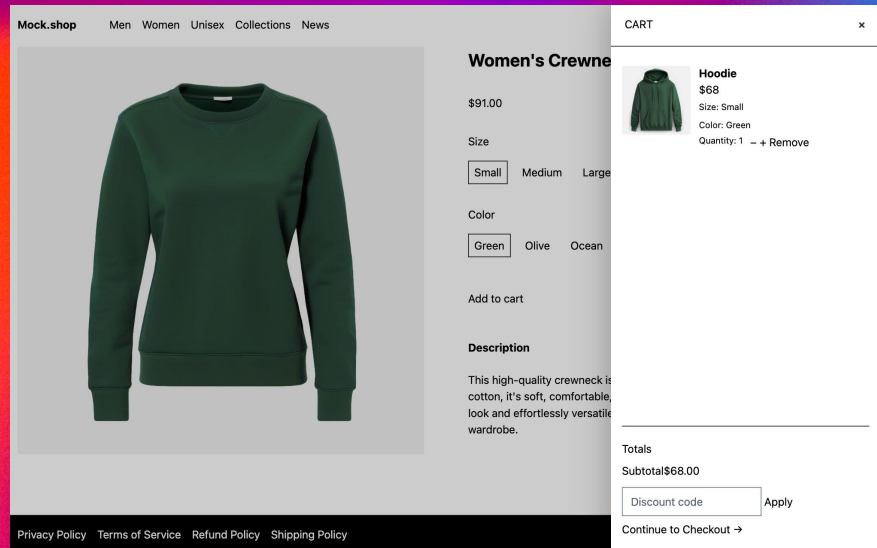
- Add to cart tramite form
- Gestione azioni su cart centralizzato
- Gestione del carrello tramite cartHandler
- Minicart
- Checkout

```
function AddToCartButton({
  analytics,
  children,
  disabled,
  lines,
  onClick,
}): {
  analytics?: unknown;
  children: React.ReactNode;
  disabled?: boolean;
  lines: CartLineInput[];
  onClick?: () => void;
} {
  return (
    <CartForm route="/cart" inputs={{lines}} action={CartForm.ACTIONS.LinesAdd}>
      {(fetcher: FetcherWithComponents<any>) => {
        <input
```



Cosa abbiamo visto

- Loader function pdp
- Query recupero product e variants
- Componente Image Hydrogen
- Dettaglio prodotto
- Visualizzazione e selezione variante
- Add to cart button
- Cart e Mini cart
- Shopify checkout



Risorse

- Hydrogen: <https://shopify.dev/docs/api/hydrogen>
- Image component:
<https://shopify.dev/docs/api/hydrogen/2023-07/components/image>
- Variant selector:
<https://shopify.dev/docs/api/hydrogen/2023-07/components/variantselector>
- Cart form:
<https://shopify.dev/docs/api/hydrogen/2023-07/components/cartform>
- Cart handler:
<https://shopify.dev/docs/api/hydrogen/2023-07/utilities/createcarthandler>
- Defer: <https://remix.run/docs/en/main/utils/defer>



GRAZIE!





Metaobjects e deploy con Oxygen





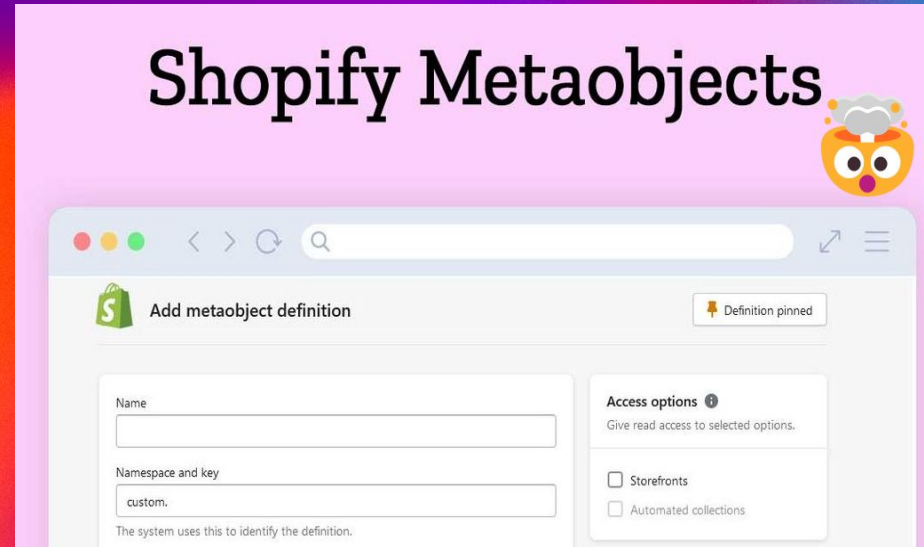
Riccardo Gai

Frontend developer

Bitforce

Metaobjects

- Oggetti personalizzati che sono utilizzati per archiviare informazioni strutturate aggiuntive al tuo negozio
- Utilizzati per conservare informazioni che non sono adatte per essere archiviate nei campi predefiniti di Shopify
- Semplificano il lavoro degli sviluppatori 😊 organizzando e strutturando le informazioni
- Prima interazione di CMS all'interno del back office



Esempio caso TooA

- Metaobject per l'home page, campi:
 - Articoli in evidenza
 - Collezioni di Robot e Card
 - Articoli
 - Prodotti in evidenza
 - Immagine di placeholder

CMS Section * Display name

home page app

highlighted-article *

tooa/trasformiamo-il-modo-di-mangiare-il-gelato-a-casa

Articolo in evidenza in home page Es: tooa/highlighted-article

artista board * Clear

 Robot - APP Change

collections cards *

 Senza Latte  Latte 2 items

articles *

tooa/born-in-italy-made-by-you • tooa/in-armonia-con-il-pianeta 2 items

highlighted-products * Clear

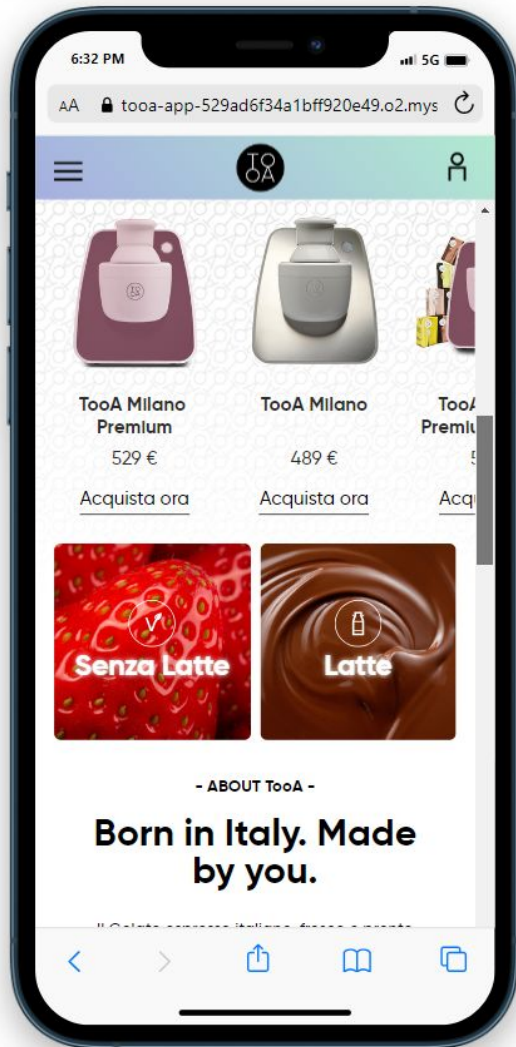
 Prodotti in evidenza - APP Change

product image placeholder * Clear

 light-gray-placeholder.png Change

Collezione
dei Robot→

Collezione delle
card→



← Articoli

← Prodotti in
evidenza



Query

- Query sul nodo metaobject
- Query sul nodo a cui le proprietà del Metaobject fanno riferimento

```
const METAOBJECT_QUERY = `#graphql
  query Metaobject(
    $country: CountryCode
    $language: LanguageCode
  ) @inContext(country: $country, language: $language) {
    metaobject(handle: { handle: "home-page", type: "home_page" }) {
      fields {
        key
        value
        type
      }
    }
  }
```

```
const COLLECTION_GRID_QUERY = (ids: string[]) => `#graphql
  query Collections(
    $country: CountryCode
    $language: LanguageCode
  ) @inContext(country: $country, language: $language) {
    ${ids}
    .map(
      (id, index) => `
      ${index} : collection(id: "${id}") {
        id
        title
        handle
        image {
          id
          url
          width
          height
          altText
        }
      }
    `
    )
  }
```

Deploy con Oxygen

- Piattaforma di hosting per hydrogen storefront
- Deploy di più ambienti con diversi environment
- Rollback immediato
- CI e CD



Come si abilita Oxigen?

- Hydrogen dallo Shopify App Store
- Shopify Github Actions
- Creazione StoreFront



Hydrogen

Price: Free

Rating (5.0)



Reviews

3

Developer

Shopify

Open app

[View demo store ›](#)

Environment e CI/CD

- Environment
 - Production
 - Custom (Staging, Development, ecc)
 - Preview
- CI/CD
 - Trigger ad ogni commit 🔥

← **Hydrogen Talk Demo** All deployments Storefront settings

Production

[main](#) [hydrogen-t...myshopify.dev](#) Public ...

 t pus:wq Current Yesterday at 1:05 pm

1 more deployment in Production

Custom environments

demo-talk [demo-talk](#) [demo-talk-...myshopify.dev](#) ...

demo commit agyb9bein...myshopify.dev 1 minute ago

View all deployments

Preview

Deployment	Preview URL	Contributor
✓ t pus:wq ↳ demo-talk (36b101a)	0x2vdfxb0...myshopify.dev	5 minutes ago
✓ t pus:wq ↳ feature/survey (36b101a)	nz7n9p9ex...myshopify.dev	Yesterday at 5:25 pm
✓ Add Shopify Oxygen deployment workflow file ↳ shopify-add-oxygen-workflow-n9gx (5c08da1)	arahr8rjh...myshopify.dev	Yesterday at 12:54 pm

View all deployments

Variabili d'ambiente

- Variabili di default per connettersi allo storefront API
- Variabili custom
- Ogni environment ha le sue variabili

← SESSION_SECRET

Key

SESSION_SECRET

Values

Value

5e282301252ecf260267cac663baaf2e8869e30b

☐ Make this value secret

Environments

Search for environments

Preview × Production ×

Value

123456789

☐ Make this value secret

Environments

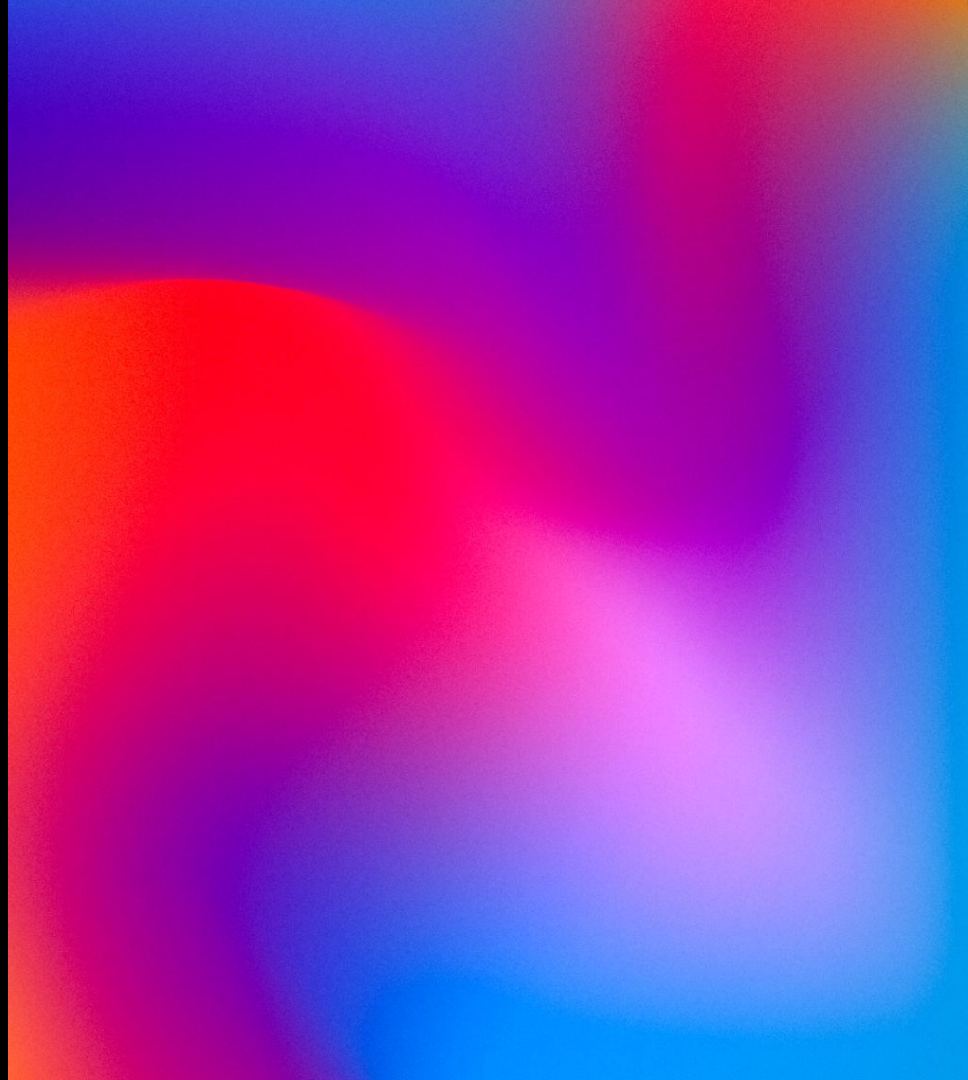
Search for environments

demo-talk ×

[+ Add another value](#)

Cosa abbiamo visto

- Metaobjects
 - Realizzare un mini CMS
 - Query per recuperare i dati
- Oxygen
 - Installazione dell'ambiente
 - Configurazione degli environments
 - Variabili ambiente
 - Rollback



Risorse

- Metaobjects

<https://shopify.dev/docs/apps/custom-data/metaobjects>

- Oxygen

<https://shopify.dev/docs/custom-storefronts/oxygen>



GRAZIE!





Coffee Break: see you in 15'

NEXT: INTEGRAZIONI



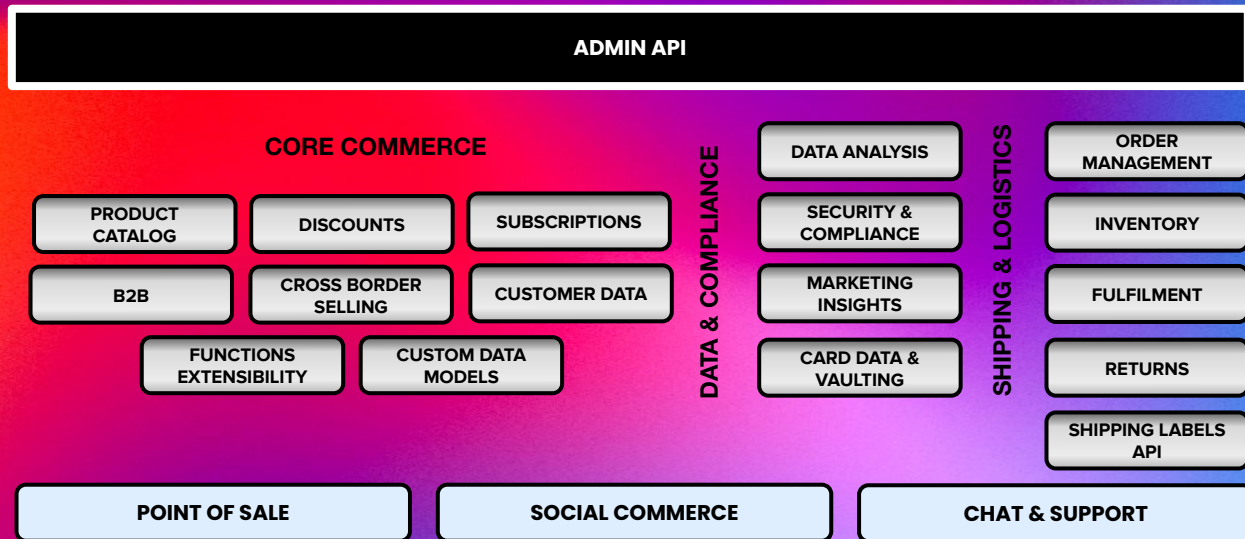
Integrazioni



Integrazioni

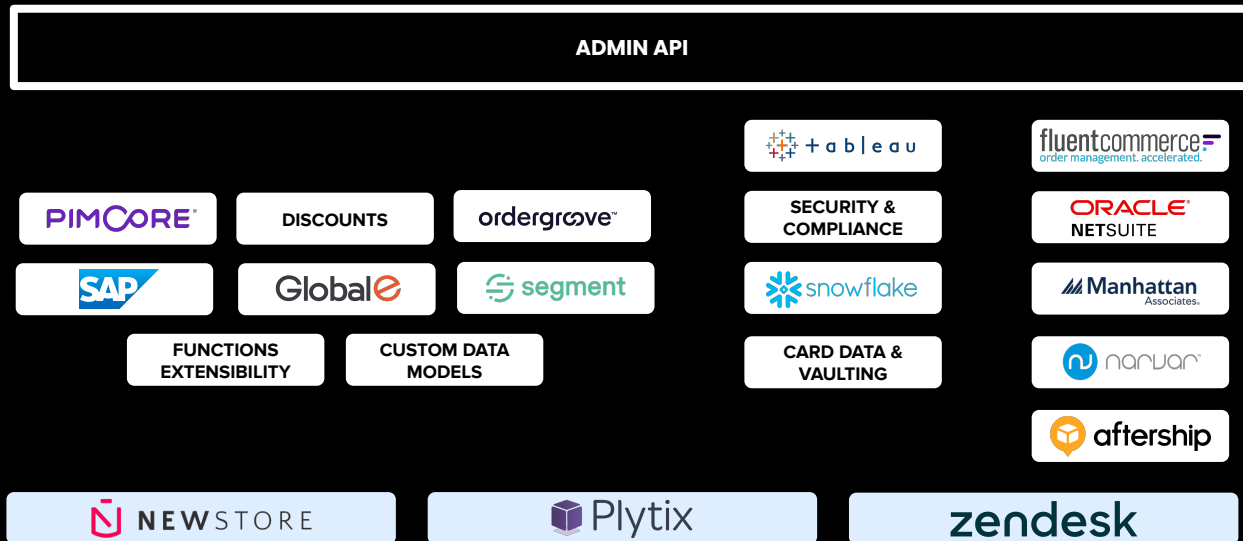
Backoffice

Integrarsi con sistemi di
terze parti



Backoffice

Integrarsi con sistemi di
terze parti



APIs

ADMIN API

- ✓ Access
- ✓ Analytics
- ✓ Apps
- ✓ B2B
- ✓ Billing
- ✓ Bulk Operations
- ✓ Collections
- ✓ Common objects
- ✓ Customers
- ✓ Discounts
- ✓ Events
- ✓ Gift card
- ✓ Inventory
- ✓ Localizations
- ✓ Marketplaces
- ✓ Metafields
- ✓ Online store
- ✓ Orders
- ✓ Payments
- ✓ Products
- ✓ Sales channel
- ✓ Shipping & fulfilment
- ✓ Store properties

STOREFRONT API

- ✓ Cart
- ✓ Checkouts
- ✓ Common objects
- ✓ Customers
- ✓ Online store
- ✓ Orders
- ✓ Products

SHOPIFY FUNCTIONS

- ✓ Order Discount API
- ✓ Product Discount API
- ✓ Shipping Discount API
- ✓ Web Assembly API

+ many more to come

PAYMENTS APP API

- ✓ App configuration
- ✓ Capture
- ✓ Common APIs
- ✓ Directives
- ✓ Payment
- ✓ Refund
- ✓ Scalars
- ✓ Void

MARKETPLACES API

MESSAGING API

MULTIPASS API

SHOPIFYQL API

Global ERP Program

Caratteristiche

- Partnership with Microsoft, Oracle NetSuite, Infor, Acumatica, and Brightpearl
- 1st party integrations maintained by the ERP vendor
- Performance, security, and support requirements for each integration



Microsoft Dynamics 365
Business Central

ORACLE
NETSUITE



Acumatica

The Cloud ERP

Brightpearl

infor



Integrare Shopify in ambienti enterprise





Luca Innocenti

Software Architect & Product Owner
Sintra

Di cosa parleremo

- **Dati e contenuti in contesto enterprise**
- **I flussi dati e gli interlocutori**
- **Impatto sul business**
- **Soluzioni ad alto livello**
- **Architettura**
- **Casi reali e soluzioni tecniche**



Shopify per tutti i target

Shopify è il primo ecommerce utilizzato sia da piccoli rivenditori che da grandi aziende.

Se per un piccolo rivenditore contano

- Rapidità di accesso
- Costo concorrenziale
- Facilità di utilizzo

Per una grande azienda

- Stabilità
- Scalabilità
- Performance



COME CAMBIANO I CONTENUTI?



CONTENUTI

Di quali contenuti parliamo?

In un mercato di prodotti intangibili
l'informazione è tutto

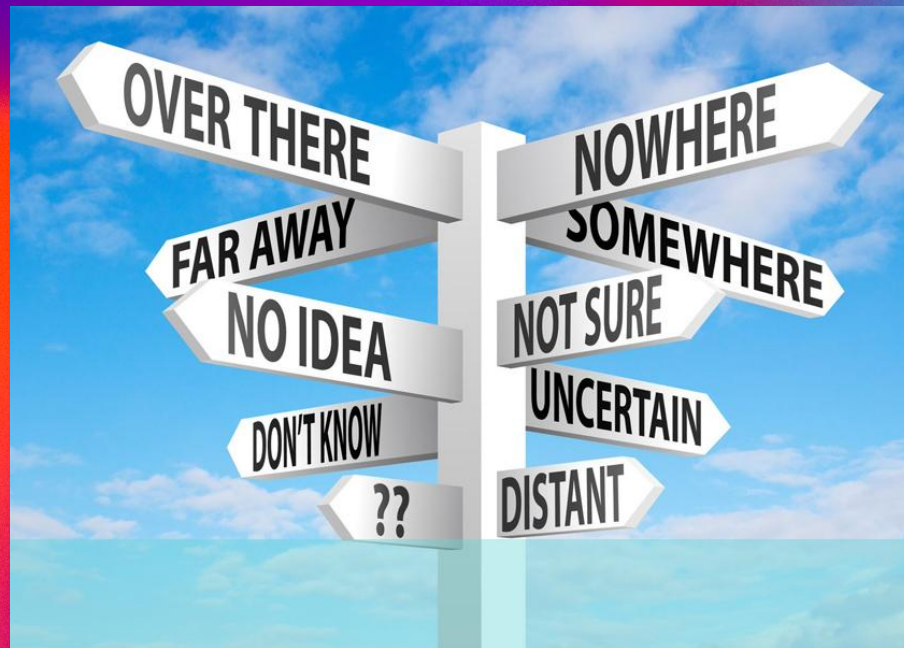
- Prodotti
- Immagini e video
- Stock
- Prezzi
- Ordini
- Fulfillment
- Customer
- ...



Da dove provengono?

Per medie realtà solitamente sono coinvolti

- ERP
- CRM
- Inserimento Manuale
- Cartelle di rete
- Excel (troppi)
- ...



Da dove provengono?

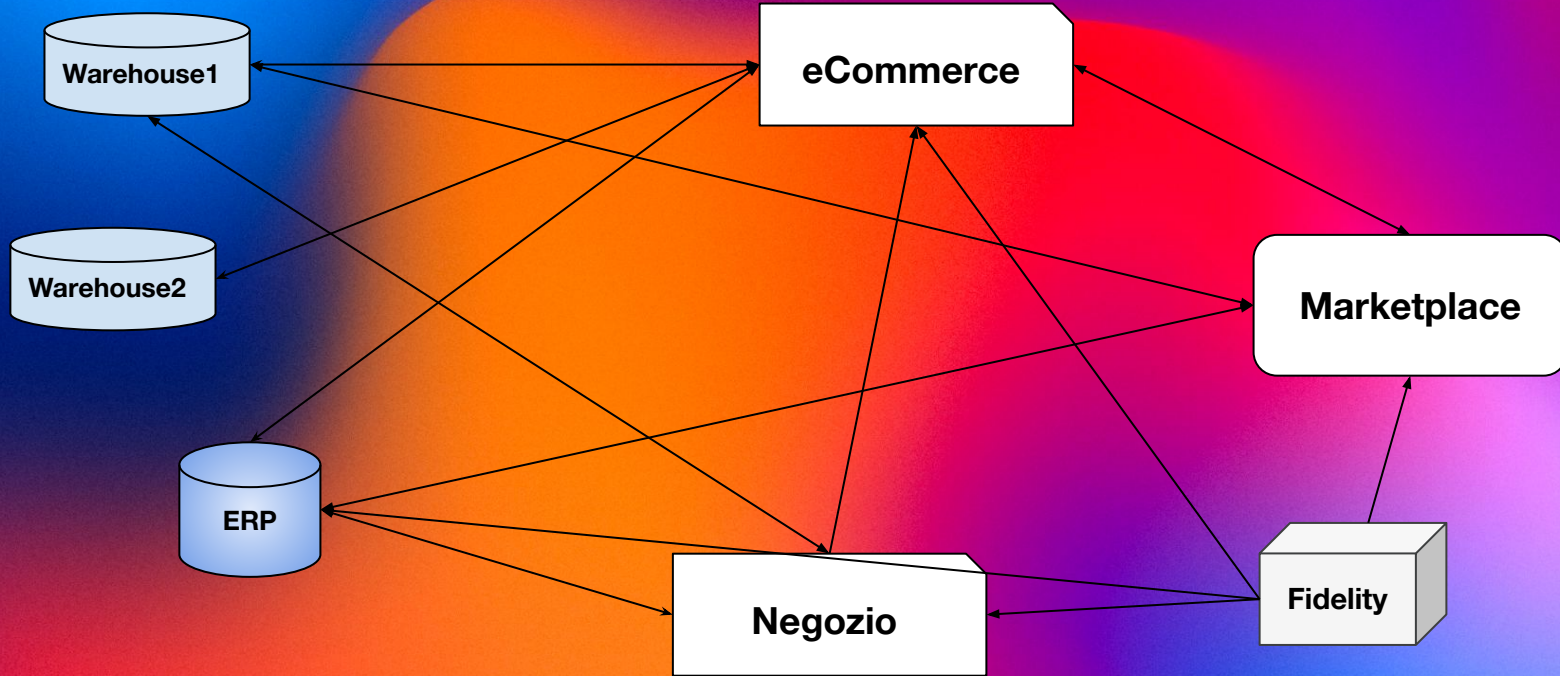
Enterprise la ownership del dato e gli interlocutori cambiano

- ERP
- CRM
- WMS
- CDP
- Cartelle di rete
- Excel (meno)
- Strumenti SEO
- Strumenti Marketing
- Traduttori
- Fidelity system
- ...

Molte sorgenti, poche certezze



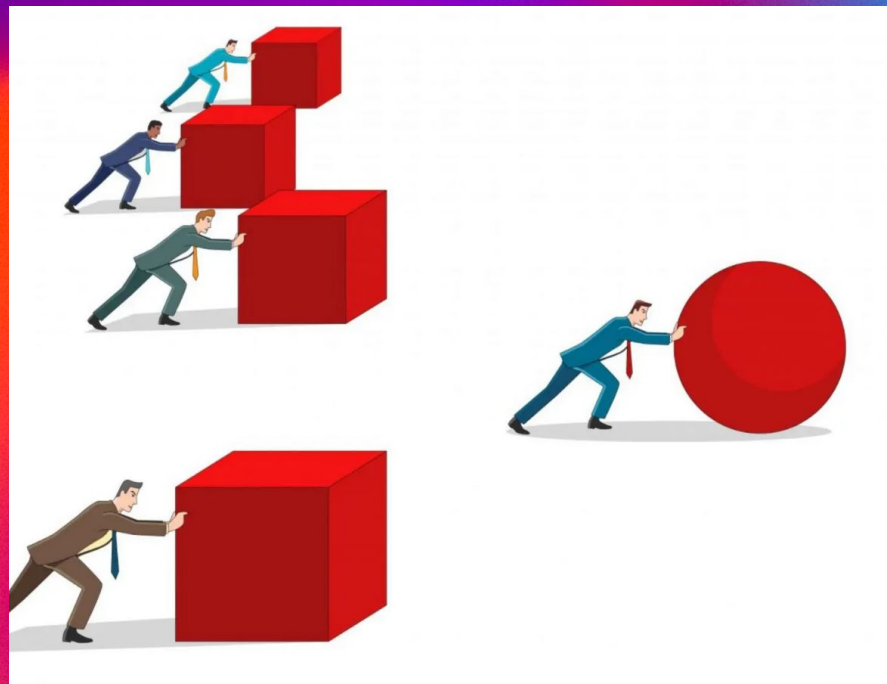
Struttura point to point



Quali sono le conseguenze di una gestione inefficace:

Cattivi contenuti, fanno cattive vendite

- Lentezza
- Ridondanza di comunicazione
- Perdita di informazioni
- Aumento dei costi di gestione
- Aumento dei costi di manutenzione
- Cliente scontenti
- Spreco degli investimenti marketing
- ...



21%

**budget marketing
sprecato a causa della
cattiva qualità dei dati**

How poor data quality impacts marketers' organizations

Respondents could select more than one response



Source: Forrester Consulting survey of 396 marketers, May 2019

COME GESTIRE DATI E CONTENUTI CON EFFICACIA?



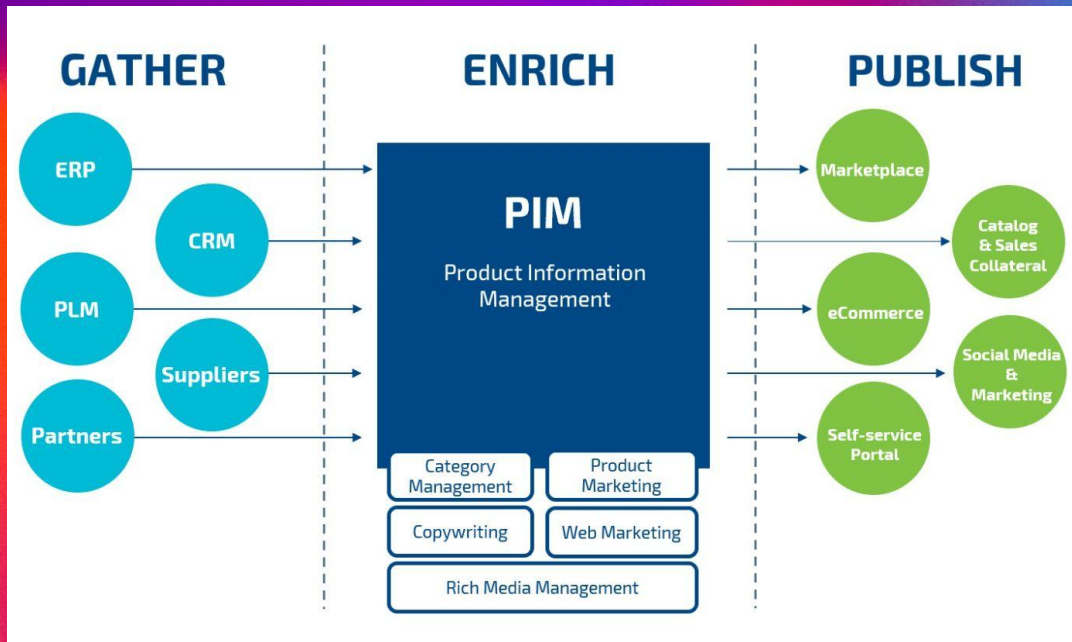
SOLUZIONE

Aggrega, arricchisci e pubblica

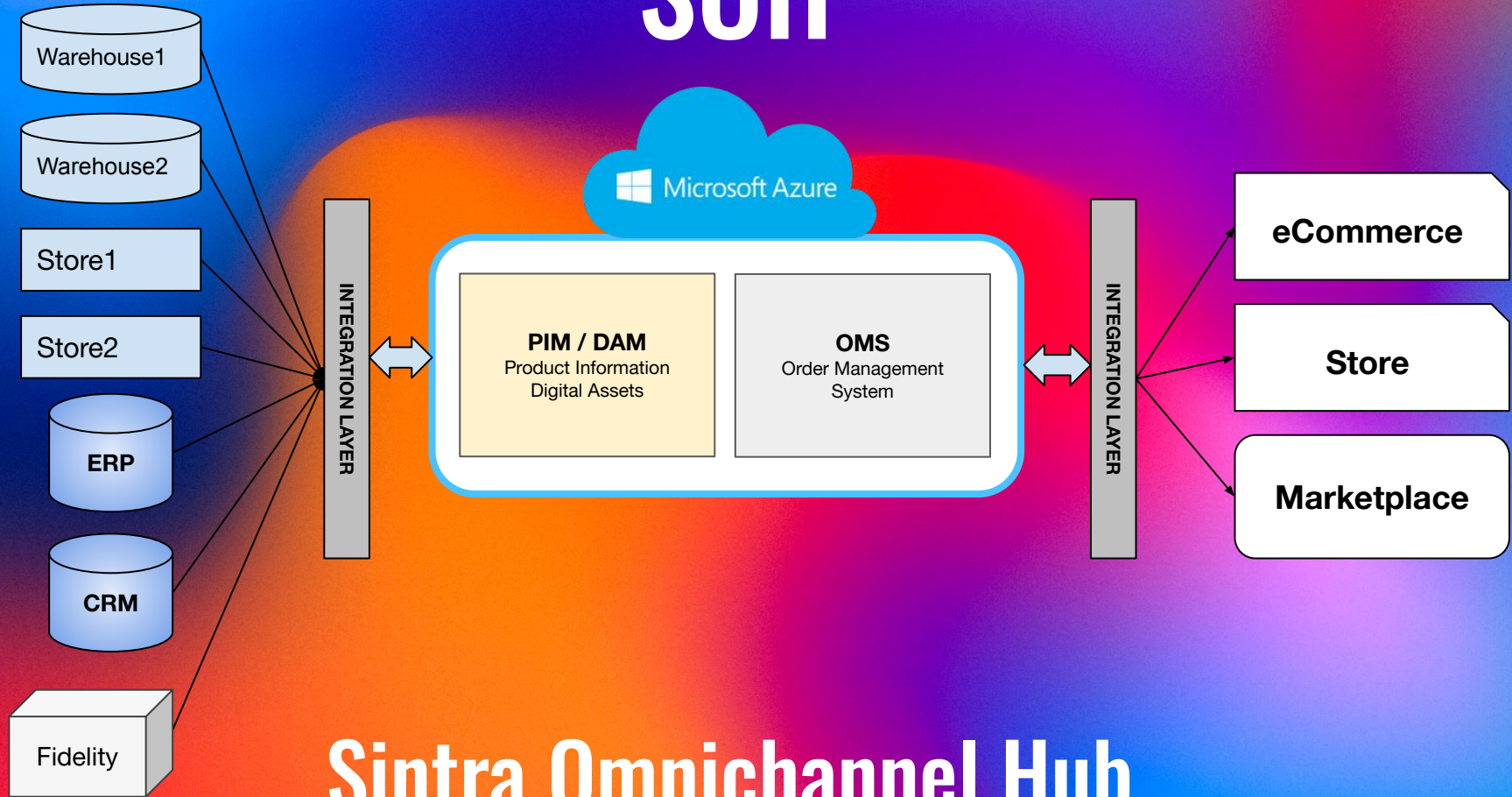
Un pattern semplice per normalizzare il dato

Vantaggi:

- Consistenza
- Disaccoppiamento
- Integrazioni accentrate
- Facilità di accesso al dato completo



SOH



Sintra Omnichannel Hub

SOH:

**Suite integrata PIM/DAM ed
OMS con plugin di integrazione
(i.e. ERP - Marketplace) per
Shopify / Shopify Plus.**

PIMCOME come PIM:

Perchè lo abbiamo scelto:

- Open Source
- Cloud e On Premise
- Leader di mercato
- PIM / DAM
- Versatile
- Versionamento del dato
- Accesso al dato profilato



PIM customizzazioni:

Le scelte che abbiamo fatto per il nostro PIM

Alcune informazioni:

- Integrato Shopify singola o multi istanza
- Integrato con i marketplace
- Multi mercato
- Multi listino (anche temporizzati)
- Controllo di completezza per canale



Listini temporizzati

Potenziamo il nostro Shopify

Scenario:

Shopify, Marketplace prezzi per mercato con la necessità di cambiare i prezzi scontati in determinate ore del giorno. Anche più volte lo stesso giorno.

Problema:

Shopify al momento non supporta questa funzionalità e l'implementazione esterna deve avere alte performance.



GraphQL per listini

GraphQL

priceListFixedPricesAdd

priceListFixedPricesDelete

Ottimizzazione flusso di sincronizzazione:

- Usato logica producer -> consumer per generare i prezzi finali
- Eliminata logica di business dal comando di invio dati
- Utilizzato GraphQL di Shopify con Chunk 250 (massimo possibile) con prestazioni maggiori

```
1 mutation priceListFixedPricesAdd($priceListId:  
  ID!, $prices: [PriceListPriceInput!]) {  
2   priceListFixedPricesAdd(priceListId:  
    $priceListId, prices: $prices) {  
3     prices {  
4       # PriceListPrice fields  
5     }  
6     userErrors {  
7       field  
8       message  
9     }  
10  }  
11 }
```

Input

Variables

Schema

```
1 {  
2   "priceListId": "",  
3   "prices": [  
4     {  
5       "compareAtPrice": {  
6         "amount": "",  
7         "currencyCode": ""  
8       },  
9       "price": {  
10        "amount": "",  
11        "currencyCode": ""  
12      },  
13      "variantId": ""  
14    }  
15  ]  
16 }
```


Risultati e tempi

Test con prodotti configurabili con media di 6 varianti

Insert/Update/Delete

Pricelist **GraphQL**

2360 Prodotti Conf/Minuto

Aggiornamento Prezzi **Base**

Api standard di update prodotto

80 Prodotti Conf/Minuto

Ambiente:



Server 4 core 8GB RAM



Account Shopify Basic

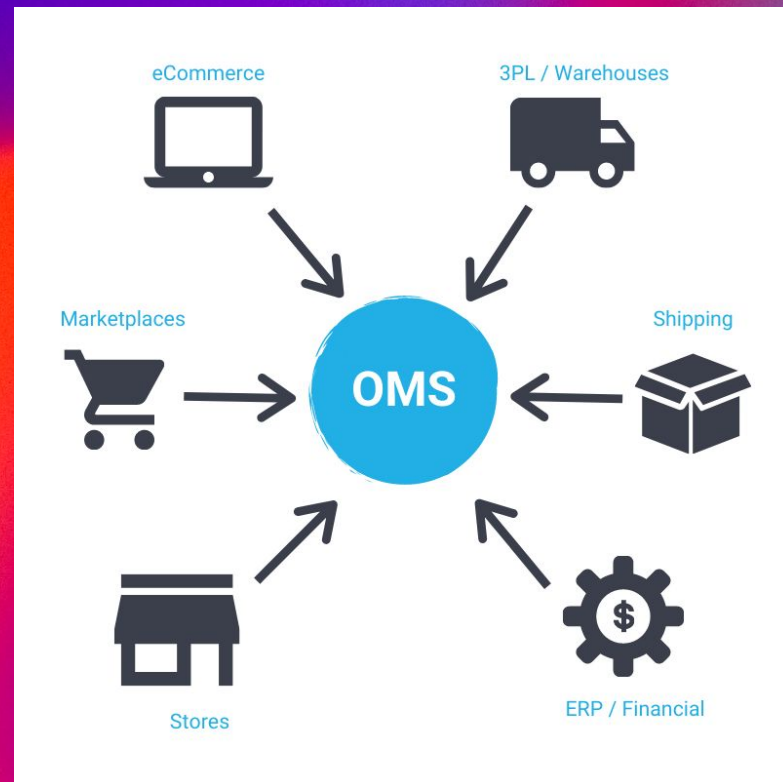


1 Target Server Pimcore

OMS Funzionalità:

Le scelte che abbiamo fatto per il nostro PIM

- Integrato con Shopify
- Integrato con Marketplace
 - Stock multi warehouse
 - Prenotazioni di stock
 - Ordini
 - Fulfillment
 - Resi e workflow di gestione
- Integrazione con POS Shopify
- Webapp di gestione OCX (BOPIS e BORIS)



Flussi OCX

Scenario:

Shopify consente Click&Collect solo se il prodotto è presente in negozio.

Il flusso di reso è complesso.

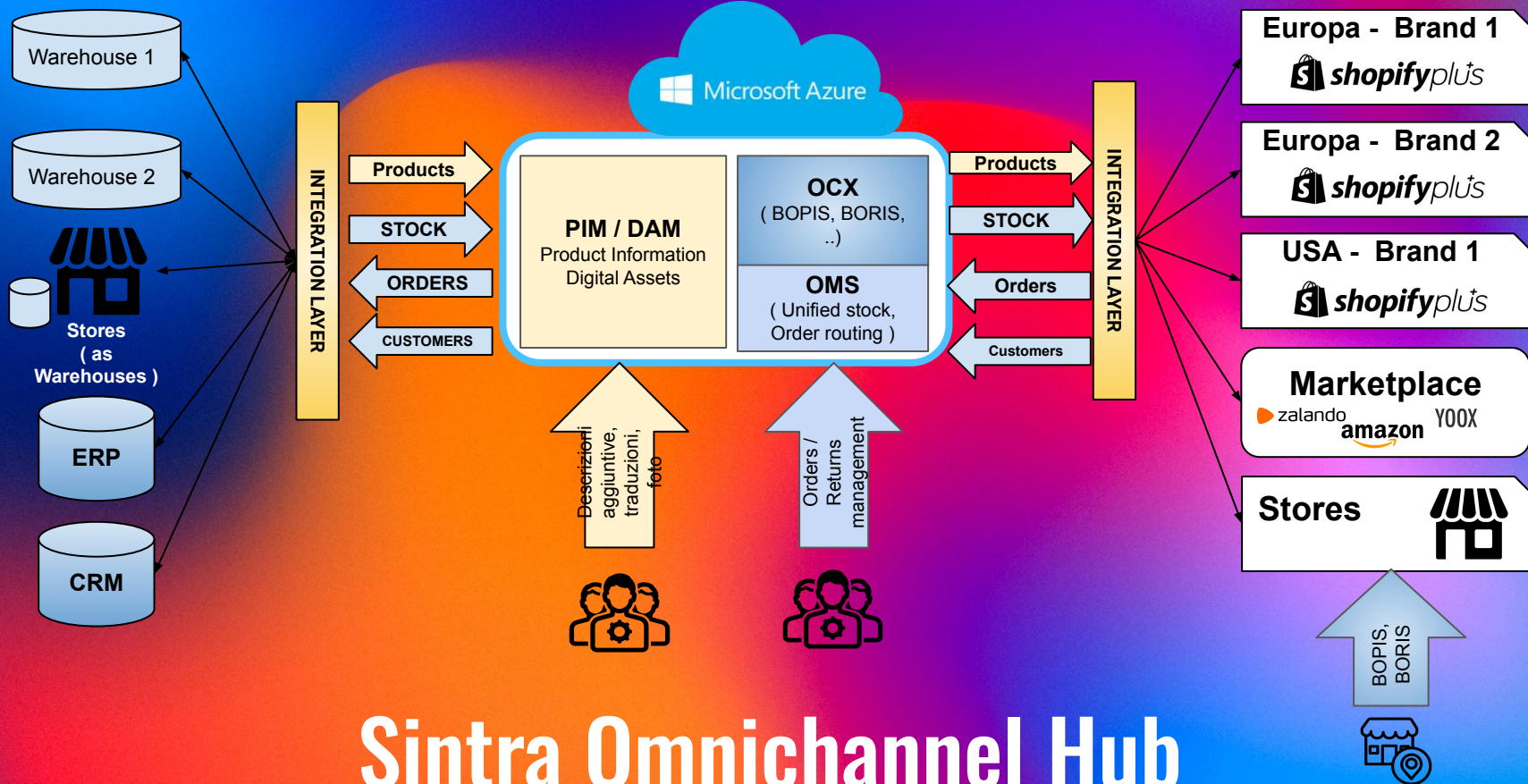
Abbiamo implementato flussi:

- BOPIS
- BORIS
- Webapp per Negozi

The screenshot shows a web application interface for managing a Click&Collect order. The top navigation bar includes a menu icon, 'SOH', 'Store Assistant', and a user profile 'wsw_milano'. The main content area displays the order details for 'Click&Collect order #1035'. It includes buttons for 'MOVE TO BE RETURNED' and 'CONFIRM PICKUP'. The order summary shows: N° order #1035, Ready to deliver at 28-08-2023, Delivered at (empty), Client Gianluca Alfредucci, N° Items 2, and Status To be delivered. The payment state is 'Completed'. Below this, an 'Items' table lists the products:

Image	SKU	Product name	Size	Color	QTY
	0135DB	6 Panel Linear Logo Strapback - Darkblue	S	Darkblue	2
	0135W	6 Panel Linear Logo Strapback - White	XL	White	1

SOH



Sintra Omnichannel Hub

Conclusioni

Shopify è un ottimo sistema per la vendita, la gestione del dato richiede però un ecosistema solido e integrato con tutti gli strumenti aziendali.

- Flussi integrazione dati
- Efficacia di comunicazione
- Sistemi Omnicanalità

Sono fondamentali per la vendita online e affrontare le sfide del presente.

La scelta è se crearseli da zero o adottare una soluzione pronta all'uso come SOH.



GRAZIE!





Nuove possibilità per Shopify B2B



LE SFIDE DEI MERCHANT

I brand sanno che il settore B2B rappresenta un'enorme opportunità per aumentare la propria crescita e raggiungere nuovi mercati, ma faticano a farlo.



Bassa flessibilità e poca personalizzazione

Crescita stagnante, lenta innovazione, vendite offline, aumento del CAC (costo di acquisizione cliente), bassa conversione e LTV (valore a vita del cliente).



Bassa efficienza e costi elevati.

Piattaforme multiple, tecnologie arcaiche, processi manuali, soluzioni legacy costose, applicazioni di terze parti.



B2B CON SHOPIFY

Una suite di potenti e flessibili funzionalità all'ingrosso integrate direttamente nel cuore di Shopify, che aiuta i commercianti Plus a personalizzare ed espandere la propria attività B2B con la potenza di Shopify.



Vendi sia B2B che DTC da un'unica dashboard



Personalizza l'esperienza di acquisto B2B.

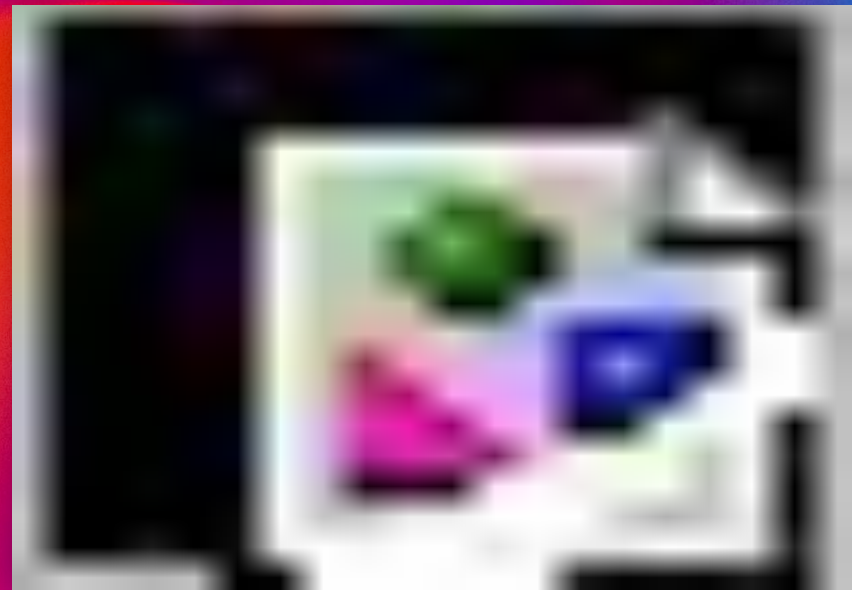


Facile da usare, completamente integrato

B2B con Shopify

Vendi all'ingrosso e direttamente da un unico admin, senza bisogno di app o sviluppo di codice.

- 01 PO numbers
- 02 Volume pricing
- 03 Wholesale migration tool
- 04 Company account requests
- 05 Quick order list
- 06 Vaulted credit cards
- 07 Storefront contextualization
- 08 Companies and locations in Flow
- 09 PayPal support



Nuove possibilità per Shopify B2B

Interlocutori diversi richiedono una comunicazione e un processo di acquisto dedicati

1.

Layout B2B per clienti B2B

- Modelli e flusso d'acquisto dedicato per il B2B

2.

Metaobjects e Landing pages

- Cosa sono i metaobjects
- Creazione di pagine e implementazioni personalizzate

3.

Estendere l'esperienza del checkout

- Inserimento di componenti e funzionalità nel checkout
- Aggiunta di logiche backend



Layout B2B per clienti B2B

Crea un'esperienza personalizzata per
tipologia di business o di mercato

- **Utilizza un tema 2.0 con la funzionalità B2B o più mercati abilitati e modifica il contenuto in base al contesto: mercato o tipologia utente**
- **Oggetto `customer.b2b?` per modificare il layout in base alla tipologia di utente**
- **Oggetto `localization.market` per recuperare il layout del mercato**
- **Oggetto `localization.country` per modificare il layout in base al paese**

```
{%- assign current_country = localization.country.iso_code | downcase -%}  
{%- case current_country -%}  
  {%- when 'at' -%}  
    {% sections 'section-at' %}  
  
  {%- when 'be' -%}  
    {% sections 'section-be' %}  
  
  {%- when 'hr' -%}  
    {% sections 'section-hr' %}  
  
  {%- when 'fr' -%}  
    {% sections 'section-fr' %}  
  
  {%- when 'de' -%}  
    {% sections 'section-de' %}  
  
{%- endcase -%}
```


Esperienza personalizzata per tipologia di utente

- Modifica delle impostazioni generali del tema (colori, font, ecc) e gestione differenziata solo di alcuni contenuti
- Soglia di spedizione gratuita differenziata per tipologia di utente e per mercato. Utilizzo a checkout delle Shopify Function API
- Inserimento del prezzo unitario in caso di confezioni con multiple unità con le diverse percentuali di IVA per mercato
- Mostrare i prodotti senza prezzi se un utente non è ancora loggato
- Mostrare la differenza di prezzo tra prezzo B2C e prezzo B2B
- Aggiunta a carrello direttamente dalla pagina collezione
- Mostrare determinati prodotti o collezioni solo per utenti B2B
- Possibilità di modifica delle email transazionali con l'aggiunta o la rimozione di informazioni dedicate



Esperienza personalizzata per Azienda e Sede

- Company, location e catalog
- Recupero delle informazioni B2B dell'utente:
 - `customer.b2b?`
 - `customer.company_available_locations`
 - `customer.current_company`
 - `customer.current_location`
 - `company`
 - `company_location`
- Inserimento di informative dedicate per Azienda e per Sede
- Facilitazione per l'utente a spostarsi tra le proprie sedi tramite un selettore aggiuntivo rispetto a quello dedicato in area account

```
1 {% if customer.b2b? %}  
2   <div>  
3     <h2>Welcome, {{ customer.name }} from {{ customer.current_company.name }}!</h2>  
4  
5     <p>You are purchasing for the {{ customer.current_location.name }} location.</p>  
6  
7     {% if customer.company_available_locations.size > 1 %}  
8       <h3>Select a different location:</h3>  
9  
10      <ul>  
11        {% for location in customer.company_available_locations %}  
12          {% unless location.current? %}  
13            <li>  
14              <a href="{{ location.url_to_set_as_current }}">{{ location.name }}</a>  
15            </li>  
16          {% endunless %}  
17        {% endfor %}  
18      </ul>  
19    {% endif %}  
20  </div>  
21 {% endif %}
```



Shopify B2B

Metaobjects e Landing pages

Layout flessibili integrati da nuove strutture dati

- **Modelli di dati personalizzati:** definizione e popolamento dei dati
- **Gestione dal pannello di amministrazione**
- **Creazione di un url dedicato**
- **Utilizzo di una struttura dati complessa per popolare un nuovo layout**

Casi d'uso

- **Partners**
- **Lavora con noi**
- **Store locator**
- **Influencers**
- **Eventi**

The screenshot displays the 'Partner' metaobject configuration page in the Shopify admin. The interface is organized into several sections:
1. **Name:** A text field containing 'Partner' with a character count of 7/255. Below it, the type is listed as 'partner'.
2. **Fields:** A list of four fields: 'Name' (partner.name), 'Description' (partner.description), 'Logo' (partner.logo), and 'Website' (partner.website). Each field has a drag handle icon on the left and a 'Display name' button on the right. An 'Add field' button is at the bottom of this section.
3. **Access:** A section with a 'Storefronts' checkbox that is checked.
4. **Features:** A section with a 'Web pages' checkbox that is checked.
5. **Web pages:** This section contains:
- 'Page title': A dropdown menu currently showing 'Name'.
- 'Meta description': A dropdown menu currently showing 'Description'.
- 'Enable in Online Store': A checked checkbox.
- 'URL handle': A text field containing 'partners'. Below this, an example URL is provided: 'https://sintra-consulting-plus.mysshopify.com/pages/partners/(entry-handle)'.
- 'Theme template': A button labeled 'Create entry to add template'.

← **Add Partner**

Name *

Stark Industries

Description

Stark Industries, a pioneering tech conglomerate, stands at the forefront of innovation. Founded by genius inventor Tony Stark, our cutting-edge advancements in aerospace, weaponry, and sustainable energy redefine industries. With a commitment to a brighter, safer future, we're driving progress worldwide. Join us in shaping a world powered by innovation and ingenuity.

Logo



Stark_Industries_Logo.png

Clear

Website

https://en.wikipedia.org/wiki/Stark_Industries

Change



Search engine listing

Stark Industries

<https://sintra-consulting-plus.myshopify.com/pages/partners/stark-industries>

Stark Industries, a pioneering tech conglomerate, stands at the forefront of innovation. Founded by genius inventor Tony Stark, our cutting-edge advancements in aerospace, weaponry, and sustainable energy redefine industries. With a commitment to a brighter, safer future, we're driving progress worldwide. Join us in sh...

Partner [View definition](#)

Status Active

Theme template
+ Create theme template

Save

← Partner	
Definition name	Metafield
PINNED (1 OF 20)	
Partners Metaobject (List) Go to definition	Select entries ⋮ Cyberdyne Systems Partner ⋮ Wayne Enterprise Partner ⋮ Stark Industries Partner

Dev Corner

Utilizzare i metaobjects da codice

- La creazione di un nuovo modello per i metaobjects genera una sottocartella nella cartella templates del tema. I files all'interno contengono la tipologia del metaobject e per i modelli aggiuntivi per quel metaobject, anche il nome del modello
- Possibilità di gestione tramite GraphQL Admin API
- Oggetti liquid accessibili dal tema:
 - metaobject_system
 - metaobject_definition
 - metaobject

Casi d'uso

- Schede tecniche per i prodotti
- Creazione pagine di riepilogo con le istanze di un metaobject

SCOPRI I NOSTRI PARTNER UFFICIALI



www.farmaciaouno.it



www.farmaciaiigea.com



www.pharmasi.it



www.redcare.it

Ingredienti

Scopri il gusto

Informazioni Nutrizionali

I preparati di ToxA contengono solo ingredienti naturali e materie prime di eccellenza.

GELATO PISTACCHIO

Valori nutrizionali per	100gr	80ml
Energie	770 kJ / 184 kcal	655 kJ / 156 kcal
Grassi	8,2 g	7,0 g
di cui acidi grassi saturi	2,3 g	1,9 g
Carboidrati	23,5 g	19,9 g
di cui zuccheri	23,5 g	19,9 g
Fibre	1,8 g	1,5 g
Proteine	5,1 g	4,3 g
Sale	0,28 g	0,24 g



Recensioni (0)

Estendere l'esperienza di acquisto

Checkout UI Extensions e Shopify Functions

- Estensione degli elementi del checkout con l'inserimento di nuovi elementi posizionabili dinamicamente
- Inserimento di nuove funzionalità nel checkout
- Personalizzazione delle logiche di backend di Shopify
- Recupero delle informazioni dallo store per implementare le proprie logiche di business

Extensions

Create extension

Development store preview ☐

Turn on

Serves work-in-progress extensions on development stores. This overrides the active app version that's live to users.

Extension Type ▾

Name ↕

Type

Last updated

[cart-checkout-validation](#)

{/} Shopify Function

Jun 20, 2023 at 10:38 PM

[italy-billing-form](#)

{/} checkout UI

Jun 8, 2023 at 3:00 PM

Learn more about [Shopify app extensions](#)

Checkout UI Extensions

- Aggiunta di informative per la gestione di ritardi nelle spedizioni o prodotti con disponibilità future
- Inserimento di campi obbligatori a checkout
- Possibilità di effettuare chiamate esterne al sito tramite la funzionalità di Network access
- Validazione, suggerimenti e modifica dell'indirizzo di spedizione: `applyShippingAddressChange`

Casi d'uso

- accettazione dei termini e delle condizioni
- campi aggiuntivi per la fatturazione elettronica
- validazione della partita iva
- Prevenzione dell'utilizzo di caratteri speciali nell'indirizzo di spedizione

Contatti Hai un account? [Accedi](#)

Email o numero di telefono cellulare
f.campiol@sintraconsulting.eu

☐ Inviarmi email con notizie e offerte

Indirizzo di spedizione

Paese / Regione
Italia

Nome (facoltativo)
Fabiola

Cognome
Campioli

Indirizzo
via fratelli lumiere 19

Interno, scala, ecc. (facoltativo)

CAP
52100

Città
Arezzo

Provincia
Arezzo

☒ Richiedi fattura
☐ Privato
☒ Azienda

Azienda - Ragione Sociale

Codice fiscale

Partita Iva

Codice SDI

PEC

☒ Richiedi fattura
☐ Privato
☒ Azienda

Azienda - Ragione Sociale

Nome compagnia richiesto

Codice fiscale

Richiesta partita IVA o codice fiscale

Partita Iva

Richiesta partita IVA o codice fiscale

Codice SDI

Richiesta PEC o SDI

PEC

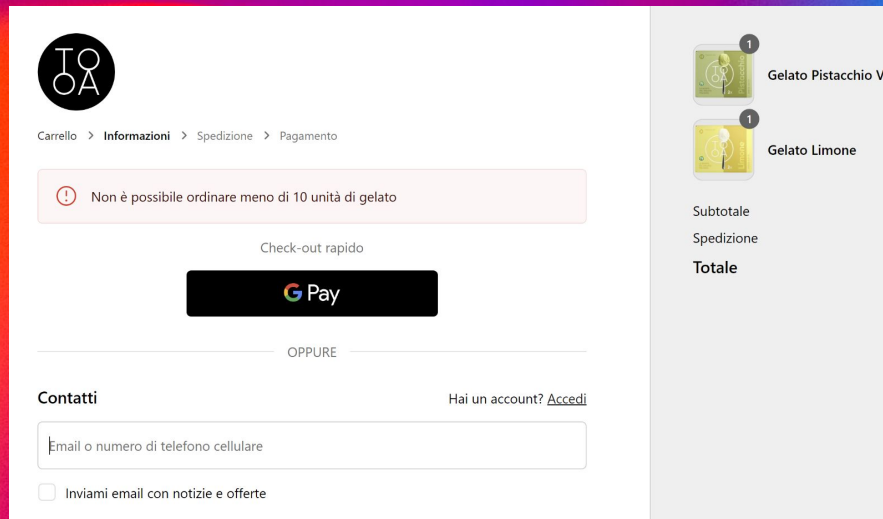
Richiesta PEC o SDI

Shopify Function API

- **Cart and Checkout Validation API in full release**
- **Cart Transform API per unificare o dividere il layout dei prodotti per i bundles in full release**
- **Discount APIs per l'integrazione di logiche di sconto inserite precedentemente tramite Script Editor (order, product e in developer preview shipping)**

Casi d'uso

- **Verifica della composizione dell'ordine per evitare che l'utente non segua le logiche di acquisto necessarie**
- **Estensione delle regole per l'acquisto di quantità tra prodotti della stessa tipologia**
- **Diversificare le regole sulla quantità di acquisto tra clienti di tipologie diverse**



Integrazione tra più stores Shopify Plus

Distribution

Select a distribution method



Public distribution

Publish your app on the Shopify App Store as listed or unlisted. Reach a global merchant audience with unlimited installs.



Custom distribution

Generate custom install links for one store or one Plus organization. Installs are limited to one store or the number of stores that an organization has.



Select

Distribution

Custom distribution

Store domain

admin.shopify.com/store/sintra-consulting-plus

For example: shop1.myshopify.com or admin.shopify.com/store/shop1

☒ Allow multi-store install for one Plus organization

This app can be installed on stores that belong to the same Plus organization as the store provided.

Generate link

Distribution

Custom distribution

Share your install link with users. They can install the app on any stores that belong to the same Plus organization as **admin.shopify.com/store/sintra-consulting-plus**.

Install link

https://admin.shopify.com/oauth/install_custom_app?client_id=eb121744180:

Copy

Expires on Wednesday, September 20, 2023 at 05:56 AM UTC.

☒ Allow multi-store install for one Plus organization

This app can be installed on stores that belong to the same Plus organization as the store provided.

GRAZIE!

